

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of)	
)	
Akira MURAKAWA)	Group Art Unit: Unassigned
)	
Application No.: Unassigned)	Examiner: Unassigned
)	
Filed: September 29, 2003)	Confirmation No.: Unassigned
)	
For: COMMUNICATION SYSTEM AND)	
METHOD IN PUBLIC KEY)	
INFRASTRUCTURE)	

CLAIM FOR CONVENTION PRIORITY

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

The benefit of the filing date of the following prior foreign application in the following foreign country is hereby requested, and the right of priority provided in 35 U.S.C. § 119 is hereby claimed:

Japanese Patent Application No. 2003-167691

Filed: June 12, 2003

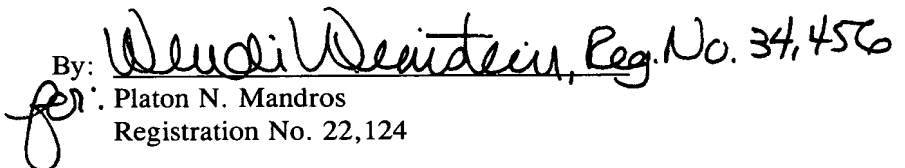
In support of this claim, enclosed is a certified copy of said prior foreign application. Said prior foreign application was referred to in the oath or declaration. Acknowledgment of receipt of the certified copy is requested.

Respectfully submitted,

BURNS, DOANE, SWECKER & MATHIS, L.L.P.

Date: September 29, 2003

By:


Platon N. Mandros
Registration No. 22,124

P.O. Box 1404
Alexandria, Virginia 22313-1404
(703) 836-6620

日 本 国 特 許 庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日

Date of Application:

2003年 6月12日

出 願 番 号

Application Number:

特願2003-167691

[ST.10/C]:

[JP2003-167691]

出 願 人

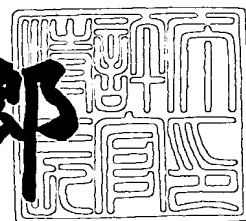
Applicant(s):

ミノルタ株式会社

2003年 6月26日

特 許 庁 長 官
Commissioner,
Japan Patent Office

太田信一郎



出証番号 出証特2003-3050737

【書類名】 特許願

【整理番号】 189480

【提出日】 平成15年 6月12日

【あて先】 特許庁長官殿

【国際特許分類】 G09C 1/00

【発明者】

【住所又は居所】 大阪府大阪市中央区安土町二丁目3番13号大阪国際ビ
ル ミノルタ株式会社内

【氏名】 村川 彰

【特許出願人】

【識別番号】 000006079

【住所又は居所】 大阪府大阪市中央区安土町二丁目3番13号大阪国際ビ
ル

【氏名又は名称】 ミノルタ株式会社

【代理人】

【識別番号】 100086405

【弁理士】

【氏名又は名称】 河宮 治

【選任した代理人】

【識別番号】 100098280

【弁理士】

【氏名又は名称】 石野 正弘

【手数料の表示】

【予納台帳番号】 163028

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

特 2 0 0 3 - 1 6 7 6 9 1

【包括委任状番号】 0113154

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 通信システムおよび方法

【特許請求の範囲】

【請求項 1】 デバイスとクライアントとがネットワークを介して通信する通信システムにおいて、

前記のデバイスは、生成された公開鍵と秘密鍵の対の中の公開鍵を含み、秘密鍵で署名されたルート証明書を保持する第 1 記憶手段と、

ルート証明書を上位認証局であることを含む証明書として作成し、作成した証明書を秘密鍵を用いて署名する証明書作成手段と、

証明書作成手段により作成された証明書をクライアントに送信する送信手段とを備え、

前記のクライアントは、前記の第 1 記憶手段に記憶されているルート証明書を記憶する第 2 記憶手段と、

前記のデバイスから受け取った証明書の署名を公開鍵を用いて検証する検証手段とを備える

通信システム。

【請求項 2】 デバイスとクライアントとがネットワークを介して通信する通信システムにおいて、

前記のデバイスは、生成された公開鍵と秘密鍵の対の中の公開鍵を含み、秘密鍵で署名されたルート証明書を保持し、

前記のクライアントは、前記のルート証明書を記憶し、

前記のデバイスは、ルート証明書を上位認証局であることを含む証明書として作成し、作成した証明書を秘密鍵を用いて署名し、作成された証明書をクライアントに送信し、

前記のクライアントは、前記のデバイスから受け取った証明書の署名を公開鍵を用いて検証する

通信方法。

【請求項 3】 前記のクライアントは、

プリンタドライバソフトをデバイスからインストールしたとき、次に、デバイ

スに対してルート証明書を要求し、

デバイスからルート証明書を受け取ると、受け取ったルート証明書を規定のフォーマットに変換し、

変換されたルート証明書をインストールすることにより、前記のルート証明書を記憶する

ことを特徴とする請求項 2 に記載された通信方法。

【請求項 4】 プリンタドライバソフトをデバイスからインストールしたとき、デバイスに対してルート証明書を要求するステップと、

デバイスからルート証明書を受け取ると、受け取ったルート証明書を規定のフォーマットに変換するステップと、

変換されたルート証明書をインストールするステップと
を有することを特徴とするコンピュータで動作可能なプログラム。

【発明の詳細な説明】

【 0 0 0 1 】

【発明の属する技術分野】

本発明は、ネットワーク環境における公開鍵を用いたデータ通信に関するものである。

【 0 0 0 2 】

【従来の技術】

公開鍵暗号方式では、公開鍵と秘密鍵の対を生成し、秘密鍵を用いてデータを暗号化し、送信する。送信先では、公開鍵を用いて復号化する。公開鍵基盤 (PKI) では、公開鍵暗号方式を用いるとともに、送信元の身元を保証するための電子証明書 (以下、単に証明書という) を発行する第 3 者機関である認証局が設けられる。証明書を復号するための公開鍵は認証局を通して受け渡しが行われる。データを送信するとき、データの送信元では、データについてのハッシュ値を作成し、データとハッシュ値を秘密鍵で暗号化して電子署名を行う。そして、公開鍵について認証局に対して電子証明書の発行を申請する。認証局では、公開鍵と秘密鍵の対を生成して、電子証明書を発行する。電子証明書では、送信元の公開鍵を含む情報が認証局の秘密鍵で暗号化されている。送信元は、ハッシュ値を用

いてデータに対して電子署名をし、認証局が発行した証明書を加えて、通信相手に送る。送信先では、認証局から認証局の公開鍵を取得して電子証明書を復号化して、送信元の公開鍵を取り出す。これにより電子署名の送信元を確認できるので、セキュリティが高められる。また、送信元の公開鍵を用いてデータとハッシュ値を復号化する。ハッシュ値が復号化できることによりデータ改竄の有無を確認できる。また、データを復号化できることにより、送信元が送信したものであることが確認できる。なお、ユーザーは、生成した公開鍵ごとに認証局に証明書の発行を要求する。証明書の発行を簡素化するデータ通信システムが、たとえば特開2001-320356号公報に記載されている。

【0003】

公開鍵基盤では、複数の認証局が、階層構造を持って組織されている。最も上位の認証局（複数）をルート認証局という。電子証明書は、階層構造をなす一連の認証局が順次署名を行ったものである。認証局の公開鍵の検証には、より上位の認証局の証明書を用いる。したがって、証明書の検証には、証明書のチェーンすなわちルート認証局までの全リストを入手する必要がある。

【0004】

【特許文献1】

特開2001-320356号公報

【0005】

【発明が解決しようとする課題】

近年、ネットワーク環境において、SSL（Secure Sockets Layer）に代表されるようなセキュアな暗号化通信の要求がある。SSLは、ウェブサーバとウェブブラウザとの間でデータを暗号化して送受信するための通信プロトコルであり、公開鍵暗号化方式により電子証明書を使って安全にデータを送受信する。SSLなどで安全に通信を行うには、サーバー側に証明書が必要である。サーバーの証明書は、外部の認証局の証明書発行サービスを有料で購入できる。しかし、イントラネットなどでは、ユーザはSSL通信だけのために外部から高額な証明書を購入したくない。一方、証明書はサーバーが作成してもよい。しかし、この場合、認証局でないものが作成した証明書であるため、SSL通信開始時に、保証

されていない証明書であるとの警告のダイアログを表示されてしまう。これはルート認証局までの証明書のリストがそろっていないためである。

【 0 0 0 6 】

この発明の目的は、デバイスとクライアントとがネットワークを介して通信するとき、ネットワークの外部から証明書を購入することなく、保証された証明書として使うことができるようにすることである。

【 0 0 0 7 】

【課題を解決するための手段】

本発明に係る通信システムは、デバイス（プリンタなど）とクライアント（PCなど）とがネットワークを介して通信する通信システムである。前記のデバイスは、生成された公開鍵と秘密鍵の対の中の公開鍵を含み、秘密鍵で署名されたルート証明書を保持する第1記憶手段と、ルート証明書を上位認証局であることを含む証明書として作成し、作成した証明書を秘密鍵を用いて署名する証明書作成手段と、証明書作成手段により作成された証明書をクライアントに送信する送信手段とを備える。また、前記のクライアントは、前記の第1記憶手段に記憶されているルート証明書を記憶する第2記憶手段と、前記のデバイスから受け取った証明書の署名を公開鍵を用いて検証する検証手段とを備える。

【 0 0 0 8 】

本発明に係る通信方法では、デバイスとクライアントとがネットワークを介して通信する通信システムにおいて、前記のデバイスは、生成された公開鍵と秘密鍵の対の中の公開鍵を含み、秘密鍵で署名されたルート証明書を保持し、前記のクライアントは、前記のルート証明書を記憶する。前記のデバイスは、ルート証明書を上位認証局であることを含む証明書として作成し、作成した証明書を秘密鍵を用いて署名し、作成された証明書をクライアントに送信し、前記のクライアントは、前記のデバイスから受け取った証明書の署名を公開鍵を用いて検証する。

【 0 0 0 9 】

好ましくは前記の方法において、前記のクライアントは、プリンタドライバソフトをデバイスからインストールしたとき、次に、デバイスに対してルート証明

書を要求し、デバイスからルート証明書を受け取ると、受け取ったルート証明書を規定のフォーマットに変換し、変換されたルート証明書をインストールすることにより、前記のルート証明書を記憶することを特徴とする。

【 0 0 1 0 】

本発明に係るコンピュータで動作可能なプログラムは、プリンタドライバソフトをデバイスからインストールしたとき、デバイスに対してルート証明書を要求するステップと、デバイスからルート証明書を受け取ると、受け取ったルート証明書を規定のフォーマットに変換するステップと、変換されたルート証明書をインストールするステップとからなることを特徴とする。

【 0 0 1 1 】

【発明の実施の形態】

以下、添付の図面を参照して本発明の実施の形態を説明する。なお、図面において、同じ参照記号は同一または同等のものを示す。

図 1 は、ネットワークを介してデータを通信するデータ通信システムを示す。プリンタ、複合装置 (MFP) などのデバイス 100 と、複数のクライアント 200 が、LAN、イントラネットなどのネットワーク 300 に接続されている。このネットワーク 300 において、通信プロトコルとして SSL (Secure Sockets Layer) を用いる。デバイス 100 はサーバーとして動作し、SSL をサポートしているウェブサーバー 120 を備える。クライアント 200 は、たとえば、パーソナルコンピュータ (PC) であり、SSL をサポートしているウェブブラウザ 216 (以下、単にブラウザともいう) を備える。デバイス 100 のウェブサーバー 120 とクライアント 200 のブラウザ 216 は、SSL でデータを交換できる。なお、図 1 に示すネットワーク 300 には、デバイス 100 とクライアント 200 がそれぞれ 1 台接続されているが、一般に、複数のデバイス 100 とクライアント 200 が接続できる。

【 0 0 1 2 】

図 2 は、デバイス 100 の 1 つである複合装置 (MFP) の内部構成を示す。複合装置は、画像読み取りを行うスキャナ部 102、画像の印刷を行うプリント部 (プリントエンジン) 104、ネットワーク 2 を介する通信を行う通信部 10

6、及び、操作指示と表示のための操作パネル108を備える。また、全体を制御するCPU110は、内部バス112を介して、前記の各部102～108に加えて、RAM114、ROM116、記憶装置（ハードディスク装置など）118、スキャナ制御部128、プリントコントローラ130に接続される。スキャナ制御部128はスキャナ部102を制御し、プリントコントローラ130はプリント部104を制御する。この複合装置は、プリンタ、複写機、スキャナなどとして機能する。また、記憶装置118には、SSLをサポートするウェブサーバー120、ルート証明書を作成する証明書作成プログラム122、自己作成証明書を作成する証明書作成プログラム124などのプログラムを記憶し、ルート証明書128、自己作成証明書130、通信されるファイル132などのデータを記憶できる。

【0013】

また、図3は、クライアント200であるパーソナルコンピュータの構成を示す。パーソナルコンピュータは、全体を制御するCPU202と、それに接続されるRAM204、ROM206を備える。CPU202には、さらに、ディスプレイ装置208と、キーボード、マウスなどの入力装置210と、外部のネットワークとの通信を行う通信装置212が接続される。CPU202には、さらに、プログラムとファイルを記憶するハードディスクを備えるハードディスク装置（HDD）214や、コンパクトディスク（CD）224aとアクセスするCD装置224が接続される。ハードディスク、コンパクトディスクなどの記憶媒体には、OSなどの他、SSLをサポートするウェブブラウザ216、プリンタドライバ218、ルート証明書をクライアントにインストールするインストールプログラム220などのプログラムや、ルート証明書222、通信されるファイル224などのデータが記憶される。プリンタドライバ218は、デバイス100の1つである複合装置（MFP）などへ送る印刷データを作成する。

【0014】

なお、デバイス100やクライアント200において、プログラムを記憶する記憶媒体としては、記憶装置118やハードディスク装置214のハードディスクの他、フレキシブルディスクやCD224aなどの各種光ディスクなどでもよ

く、これらはそれぞれ対応する記憶装置（フレキシブルディスク装置や光ディスク装置）で利用される。

【0015】

デバイス100のウェブサーバー120とクライアント200のウェブブラウザ216がSSLでデータを送信するとき、サーバー認証、クライアント認証および通信内容の暗号化を行う。デバイス100は、ルート証明書122を保持している。ルート証明書122は、自分で作成することも、認証局に発行を依頼することもできるが、ここでは、ルート証明書122は、後で説明するようにデバイス100が作成し保持している。デバイス100が作成する場合、デバイス側でルート証明書の属性値を自由に変更できるという利点がある。ルート証明書122は、ルート証明書の作成の際に生成された公開鍵を含んでいる。一方、このルート証明書122が、あらかじめクライアント200にインストールされている。これが、クライアント200に記憶されているルート証明書220である。なおルート証明書220のインストールは、好ましくは自動的に行われる。たとえばデバイス100のためのプリンタドライバ218をクライアント200にインストールするときにインストールされる。また、インストールは、好ましくはユーザーの確認のもとに行われる。クライアント200は、ルート証明書220を保持しているので、後で説明するように、デバイス100から受け取った証明書の認証を行うときに、外部の認証局で発行された証明書を必要としない。なお、ルート証明書220をROMなどに記録してクライアント200に提供すれば、その改竄が防止できる。

【0016】

証明書は、デバイス200内のプログラム122、124によりX.509形式で作成されるが、X.509形式の証明書は以下の情報を含む。証明書のバージョン、証明書の通し番号、署名値と署名に使ったアルゴリズムとパラメータ、発行者の名前、住所など、証明書の有効期限、証明書の持主の名前、住所などおよび証明対象の公開鍵情報。プログラムの入力要求に対応してユーザーが必要な情報を入力すると、証明書がそれに基づいて作成される。X.509形式の証明書は、内部ではASN.1形式で保存するが、外部ではPKCS#12、PKCS#7などの形式で

保存する。自己作成証明書を作成する証明書作成プログラム124を用いて証明書を作成するとき、発行者の名前にルート証明書を記載する。

【0017】

SSLによる送信について説明すると、デバイス100のウェブサーバー102に秘密鍵と公開鍵があらかじめ用意されている。図4に示すように、クライアント200からデバイス100にSSL接続が要求されると、デバイス100は、使用する暗号化と圧縮のアルゴリズムを決定し、クライアント200に通知し、自己の公開鍵を含むX.509形式の証明書（自己作成証明書）を作成し、ルート認証局までの証明書のリスト（証明書チェーン）を含めて、クライアント200に渡す。クライアント200は、証明書を検証する際に、クライアント200にインストール済みのルート証明書220の公開鍵を用いて証明書を復号して検証を行い、身元が保証された証明書と判断する。なお、必要ならばクライアント認証のための通信を行う。

【0018】

認証の後に続く暗号化通信では、データ通信にセッション鍵（共通鍵）を使用する。そこで、クライアント200は、暗号化に使用するセッション鍵を生成するための情報を、デバイスの証明書に含まれる公開鍵を用いて暗号化してデバイス100に渡す。デバイス100は、証明書の秘密鍵で復号してセッション鍵を取得する。これにより、この後の通信ではアプリケーションのデータの暗号化通信が可能になる。

【0019】

図5により、証明書の取扱いについてさらに説明すると、デバイス200は、ルート証明書を保持していて、SSLで送信するとき、1対の公開鍵と秘密鍵を生成し、その公開鍵を含む自己作成証明書を作成する。証明書のチェーンは、ルート証明書と自己作成証明書の2階層からなる。ここで、図6に示すように、自己作成証明書では、上位認証局としてルート証明書を指定している。デバイス200は、電子署名のため、ルート証明書を上位認証局として含む自己作成証明書について、所定のハッシュ関数を用いてハッシュ値を求めて、署名を行い、自己作成証明書に組み込む。そして、データと自己作成証明書とをSSLでクライア

ント100に送信する。

【0020】

認証局の公開鍵の検証には、より上位の認証局の証明書を用いる。したがって、証明書の検証には、証明書のチェーンすなわちルート認証局までの全リストを入手する必要がある。クライアント100では、デバイス200から送られてきた自己作成証明書を検証する際に、上位認証局の証明書としてルート証明書が指定されているので、クライアント100にインストール済のルート証明書により検証を行い、保証された証明書と判断する。したがって、ネットワーク外の認証局から証明書を発行してもらわなくても、証明書の検証が行える。

【0021】

証明書の検証は以下のように行う。

- 1) 自己作成証明書を署名した証明書（ここでは、ルート証明書となる）を見つける。ルート証明書がクライアント100内に存在しているので、ルート証明書の存在が保証される。したがって、SSL開始時に警告が発生されることはない。
- 2) ルート証明書の公開鍵でハッシュ値を復号する。
- 3) 自己証明書のハッシュ値を求める。
- 4) 次に2)と3)の結果を比べ、ハッシュ値が同じであることを確認する。値が同じであれば、自己証明書が改竄されていないことが証明できる。

【0022】

図7は、デバイス100のCPU110によるルート証明書作成のフローチャートを示す。まず、ルート証明書の作成に必要な情報を入力する(S10)。情報は、たとえば、組織名、部署名、市町村名、都道府県名、国名、有効期限、暗号化方法などである。次に、乱数を発生させ、公開鍵と秘密鍵のペアを生成する(S12)。生成にはたとえばRSA方式を用いる。次に、MD5(Message Digest #5)などの所定のハッシュ値を求めるアルゴリズムを用いて、証明書のハッシュ値を求める(S14)。次に、ハッシュ値を秘密鍵で暗号化する(S16)。これを、秘密鍵で暗号化した証明書と組み合わせる。このように証明書に署名を付けてルート証明書とする(S18)。

【 0 0 2 3 】

図 8 は、デバイス 1 0 0 の CPU 1 1 0 による自己証明書作成のフローチャートを示す。まず、証明書の作成に必要な情報を入力する（S 2 0）。情報は、たとえば、組織名、部署名、市町村名、都道府県名、国名、有効期限、暗号化方法などである。証明書の上位パス情報および発行者の情報としてルート証明書を入れる。次に、乱数を発生させ、公開鍵と秘密鍵のペアを生成する（S 2 2）。鍵生成にはたとえば RSA 方式を用いる。次に、MD 5 などの所定のハッシュ値を求めるアルゴリズムを用いて、証明書のハッシュ値を求める（S 2 4）。次に、ハッシュ値をルート証明書の秘密鍵で暗号化する（S 2 6）。これが証明書の署名となる。次に、証明書に署名を付けて自己作成証明書とする（S 2 8）。

【 0 0 2 4 】

図 9 は、クライアント 2 0 0 の CPU 2 0 2 による、デバイス（サーバー） 1 0 0 から送付された証明書の検証のフローチャートを示す。ここでは、ブラウザにおける SSL プロトコルによる検証処理の中で、ルート証明書に関する部分だけを示す。デバイス 1 0 0 から証明書を受け取ると、検証が開始される。まず、その証明書の記載を基に上位証明書（ここではルート証明書）を得て（S 4 0）、上位の証明書の認証局（CA）が信頼できるか否かを判断する（S 4 2）。ここで、クライアント 2 0 0 側で、信頼するルート認証局として登録されている場合や、インターネットでルート認証局に問い合わせで登録されている場合は、信頼できると判断できる。ここでは、あらかじめルート証明書がインストールされて、信頼するルート認証局であることを含む証明書として登録されているので、信頼できると判断される。上位の証明書の認証局が信頼できる場合は、次に、署名をルート証明書の公開鍵で復号化する（S 4 4）。復号化できた場合は（S 4 6 で YES）、上位の認証局で証明されたと判断できる。

【 0 0 2 5 】

次に、証明書のハッシュ値（Hash2）を生成する（S 4 8）。次に、2 つのハッシュ値 Hash1、Hash2 を比較する（S 5 0）。一致する場合は、送付された証明書は改竄されていず、信頼できるので、SSL 通信処理を続行する（S 5 2）。

【 0 0 2 6 】

一方、上位の証明書の認証局が信頼できない場合や、署名の復号ができない場合や、ハッシュ値Hash1、Hash2が一致しない場合は、証明書は信頼できないので、証明書は信頼できない旨の警告ダイアログを画面に表示する（S54）。

【0027】

次に、デバイス100のクライアント（PC）200へのルート証明書220のインストールについて説明する。1つのインストール方法では、デバイス100は複合装置やプリンタであり、デバイス100へのプリンタドライバ218のインストール時に、証明書をデバイス（プリンタ）100からクライアント200に転送し、インストールする。図10は、プリンタドライバと共にインストールされる場合のルート証明書のインストールプログラム220のフローチャートを示す。クライアント200のCPU202は、まず、プリンタドライバソフトをインストールする（S100）。次に、ルート証明書のクライアント200へのインストールの確認を画面に表示する（S102）。クライアント200のユーザによるインストールの確認の入力を受け取ると（S104でYES）、次に、デバイス100に対してルート証明書を要求する（S106）。デバイス100からルート証明書を受け取ると（S108）、受け取ったルート証明書を規定のフォーマットに変換する。これにより、クライアント200へのインストールが可能になるので、変換されたルート証明書をインストールする（S110）。

【0028】

好ましくは前記の方法において、クライアントは、プリンタドライバソフトをデバイスからインストールしたとき、次に、デバイスに対してルート証明書を要求し、デバイスからルート証明書を受け取ると、受け取ったルート証明書を規定のフォーマットに変換し、変換されたルート証明書をインストールすることにより、前記のルート証明書を記憶する。

【0029】

また、別のインストール方法では、デバイス100から転送された証明書をクライアント（PC）200のハードディスク装置214にファイルとして一旦保存し、証明書を管理する証明書ダイアログからインポートする。ここで説明する例では、デバイス100から送付されたルート証明書を一旦ファイルに落とし、

証明書ダイアログによりユーザーがインストールする。図 1 1 は、ルート証明書のインストールのフローチャートを示す。クライアント 2 0 0 の CPU 2 0 2 は、まず、デバイス 1 0 0 からルート証明書を受け取る (S 2 0 0)。デバイス 1 0 0 で保持しているルート証明書 1 2 2 は、規定のフォーマットに変換することで、クライアント (PC) 2 0 0 にインストール可能となる。そこで、ルート証明書のフォーマットを変換し (S 2 0 2)、ルート証明書ファイルをハードディスク装置 2 1 4 に保存する (S 2 0 4)。次に、得られたルート証明書ファイルをインストールする (S 2 0 6)。たとえば、ウィンドウズ (登録商標) のブラウザ (Internet Explorer) の場合、証明書ダイアログには、インストールされた証明書の一覧を表示するダイアログボックスが表示される。証明書をインストールするとき、証明書ダイアログに設けられている「インポート」ボタンを押して、「信頼されたルート証明機関」に証明書をインポートすればよい。

【 0 0 3 0 】

なお、ルート証明書をインストールするためのプログラムは、たとえば、ウィンドウズ (登録商標) のオープン API の中に上述の機能 (関数) を作って、公開してもよい。

【 0 0 3 1 】

以上の説明では、認証が 1 段階である場合について説明した。より一般的に認証が複数段階である場合でも、より上位の複数の証明書をクライアント 1 0 0 にインストールしておけばよい。これにより、クライアント 1 0 0 は、ルート認証局までの証明書のチェーンを使用して検証を行える。

【 0 0 3 2 】

図 1 2 に示す例では、証明書のチェーンは、ルート証明書、中間証明書及び自己作成証明書の 3 階層からなる。デバイス 2 0 0 (プリンタ、MFP など) は、ルート証明書と中間証明書を保持していて、あらかじめ、クライアント 1 0 0 にルート証明書と中間証明書をインストールしておく。デバイス 2 0 0 は、SSL で送信するとき、1 対の公開鍵と秘密鍵を生成し、その公開鍵を含む自己作成証明書を作成する。ここで、中間証明書では、上位認証局としてルート証明書を指定していて、署名を組み込んでおり、自己作成証明書では、上位認証局として中

間証明書を指定し、署名を組み込んでいる。デバイス200は、電子署名のため、中間証明書を上位認証局として含む自己作成証明書について、所定のハッシュ関数を用いてハッシュ値を求めて、署名を行い、自己作成証明書に組み込む。そして、データと自己作成証明書とをSSLでクライアント100に送信する。クライアント100では、デバイス200から送られてきた自己作成証明書を検証する際に、インストールされている証明書のチェーンを用いる。自己作成証明書には上位認証局の証明書として中間証明書が指定されているので、インストール済の中間証明書により検証を行い、さらに、中間証明書について、インストール済みのルート証明書により検証を行う。これにより自己作成証明書を、保証された証明書と判断する。

【0033】

上述の暗号化データ通信の実施の形態は、SSLによる通信の場合について説明しているが、一般に、公開鍵暗号化方式を用い、ルート認証局までの証明書のチェーンを用いて認証を行うシステムに適用できる。そのようなシステムにおいて、サーバー装置（デバイス）にルート証明書を保持し、それにネットワークを介して接続されるクライアントに、そのルート証明書をインストールしておく。公開鍵暗号化方式でクライアントからデータを送信する際に、クライアントは、サーバー装置から送られた証明書の認証において、インストールされているルート証明書を用いる。認証の後にデータの暗号化通信を行う。実施の形態では、セッション鍵を用いてデータを通信するが、暗号化方式はこれには限られない。

【0034】

そのようなサーバー装置とクライアントとがネットワークを介して通信するデータ通信システムにおいて、サーバー装置は、生成された公開鍵と秘密鍵の対の中の公開鍵を含み、秘密鍵で署名されたルート証明書を保持する第1記憶手段と、ルート証明書を上位認証局として含む証明書を作成し、作成した証明書を秘密鍵を用いて署名する証明書作成手段と、証明書作成手段により作成された証明書をクライアントに送信する送信手段とを備える。好ましくは、サーバー装置において、第1記憶手段はROMである。これにより、ルート証明書は変更されなくなる。一方、クライアントは、前記の第1記憶手段に記憶されているルート証明

書を記憶する第2記憶手段と、前記のデバイスから受け取った証明書の署名を公開鍵を用いて検証する検証手段とを備える。

【0035】

【発明の効果】

デバイスのルート証明書をクライアントにインストールしておくので、デバイスからクライアントへのセキュアな暗号化通信の開始時に警告を発生（表示）されない。

【図面の簡単な説明】

【図1】 本発明のデータ通信システムのブロック図

【図2】 複合装置の構成を示すブロック図

【図3】 パーソナルコンピュータの構成を示すブロック図

【図4】 SSLにおけるサーバーとクライアントとの間の通信シーケンスの図

【図5】 ルート証明書と自己作成証明書の取扱いを説明するための図

【図6】 ルート証明書と自己作成証明書を用いた認証を説明するための図

【図7】 ルート証明書作成のフローチャート

【図8】 自己証明書作成のフローチャート

【図9】 証明書検証のフローチャート

【図10】 ルート証明書のインストールのフローチャート

【図11】 ファイルからの証明書のインストールのフローチャート

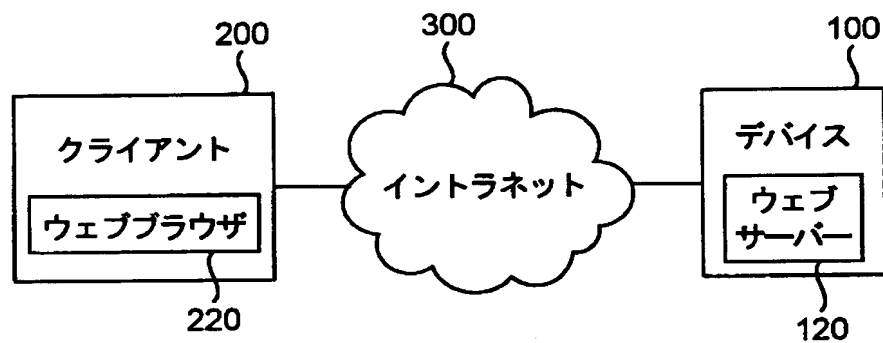
【図12】 複数段階の検証を行う例を説明するための図

【符号の説明】

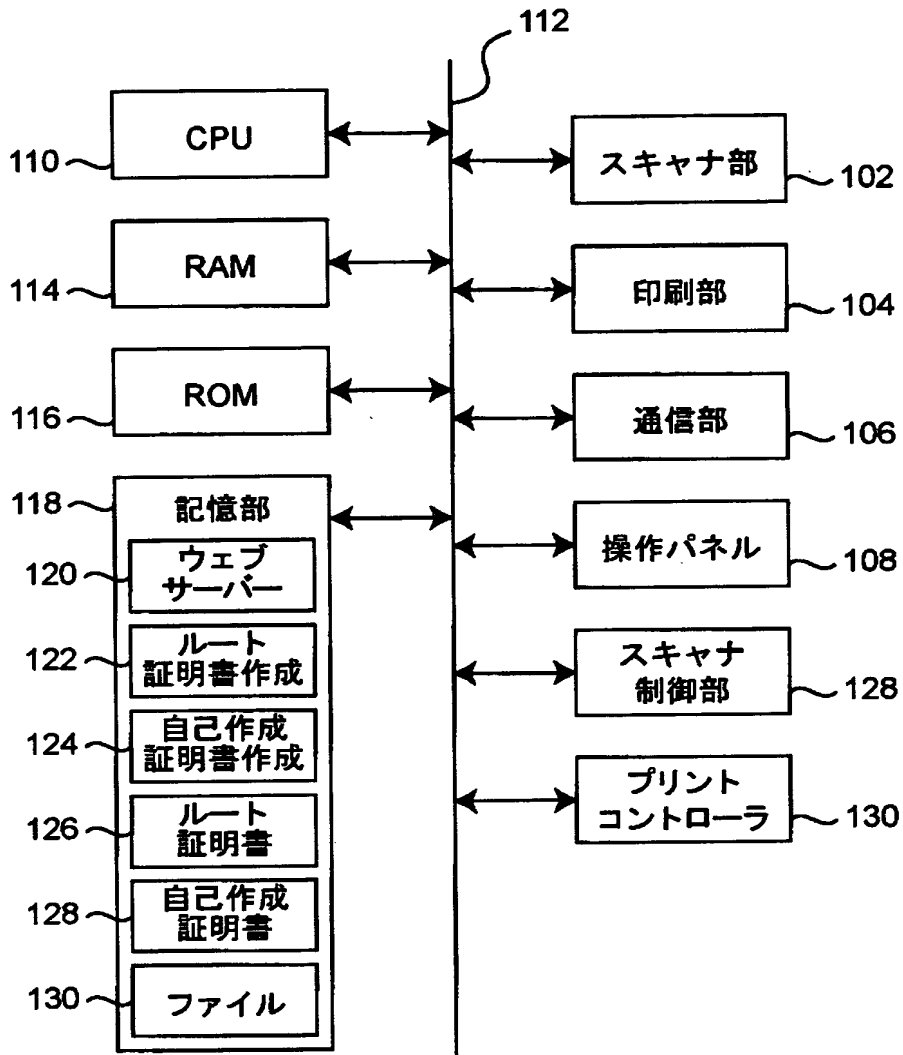
100 デバイス、 110 CPU、 116 ROM、 118 記憶装置、 120 ウェブサーバー、 122 ルート証明書を作成する証明書作成プログラム、 124 自己作成証明書を作成する証明書作成プログラム、 128 ルート証明書、 130 自己作成証明書、 200 クライアント、 202 CPU、 214 ハードディスク装置、 216 ウェブブラウザ、 218 プリンタドライバー、 220 インストールプログラム、 222 ルート証明書、 300 ネットワーク。

【書類名】 図面

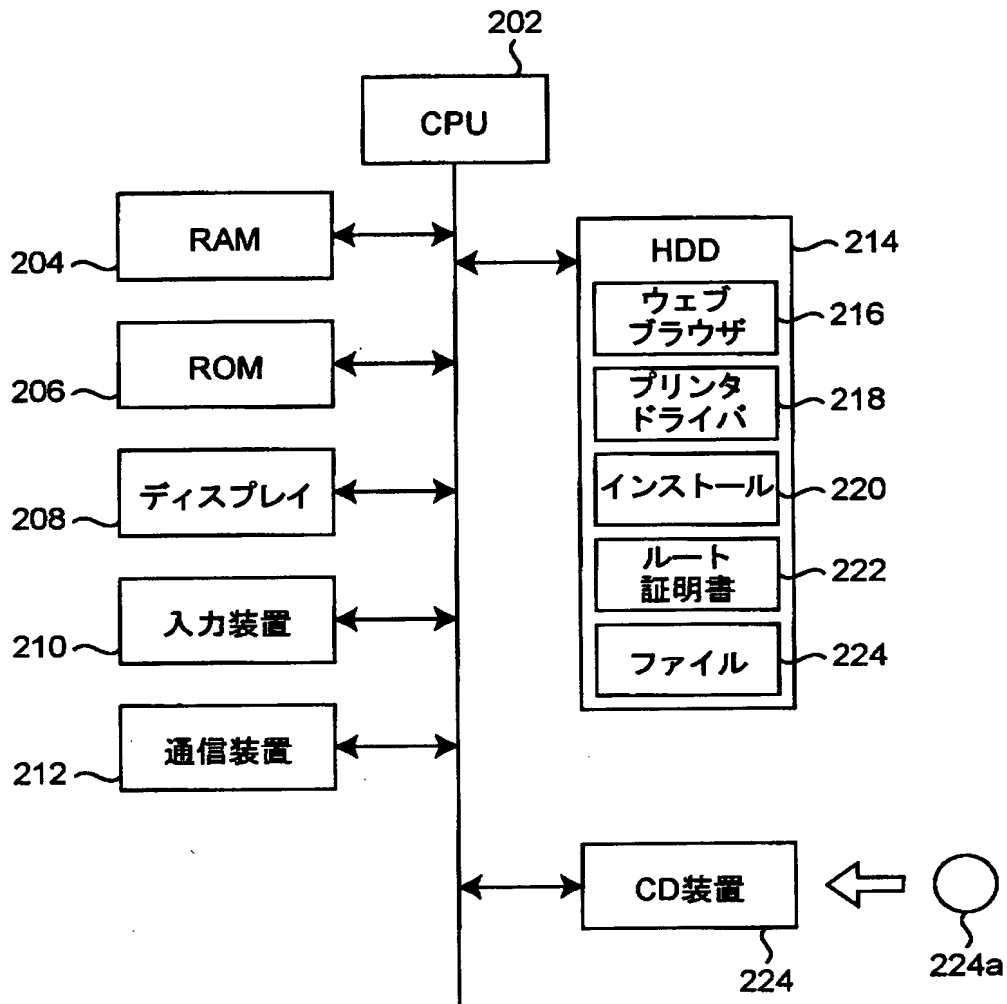
【図 1】



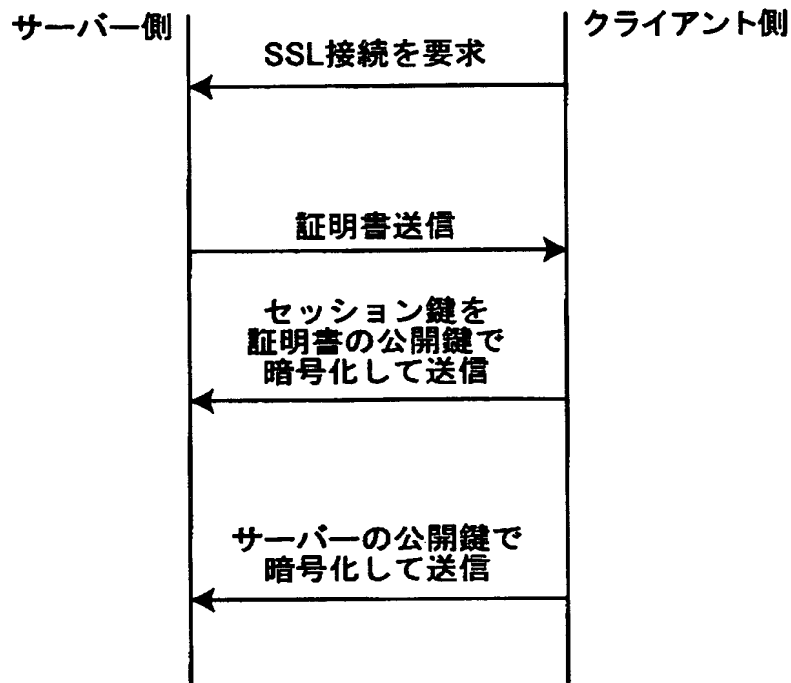
【図 2】



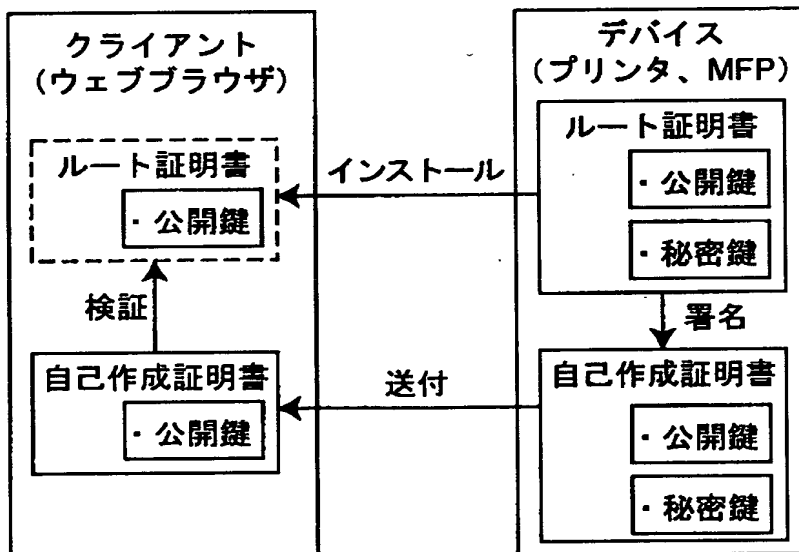
【図 3】



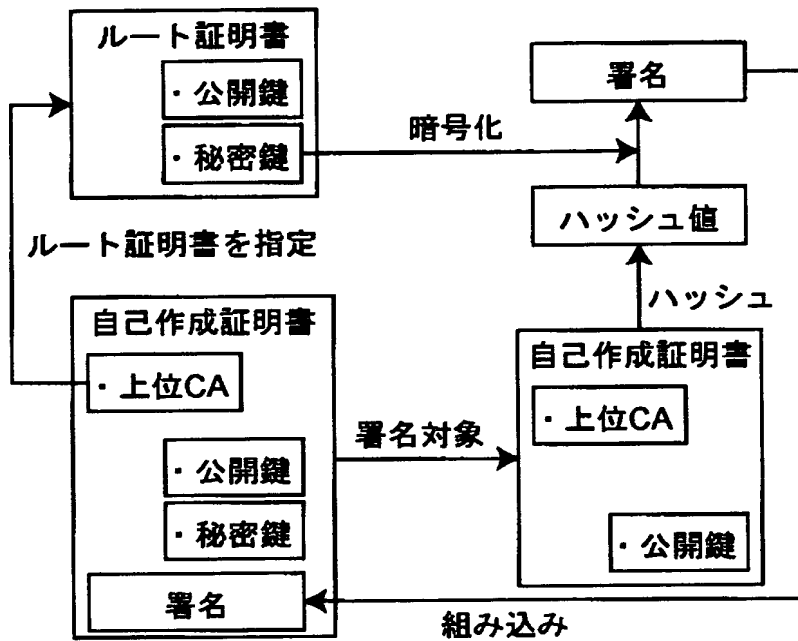
【図 4】



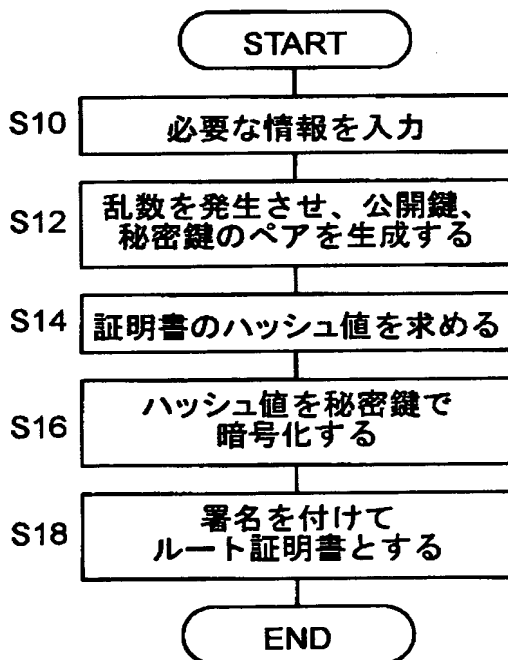
【図 5】



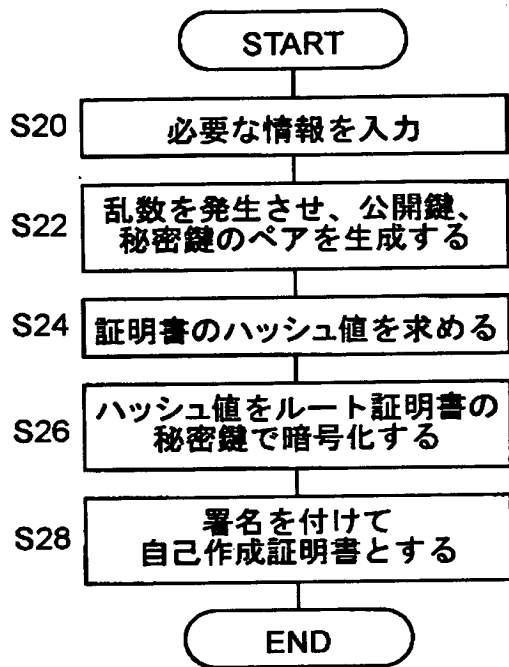
【図 6】



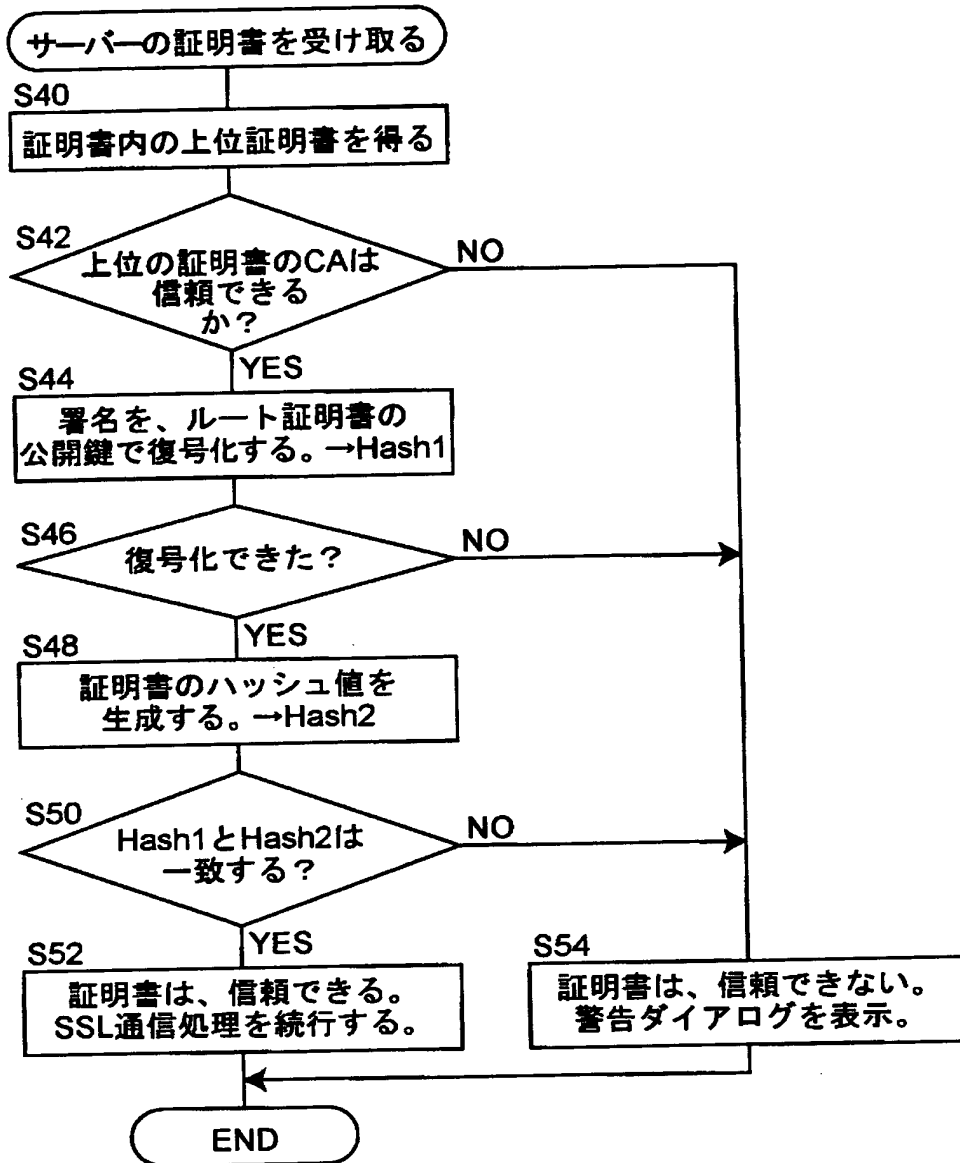
【図 7】



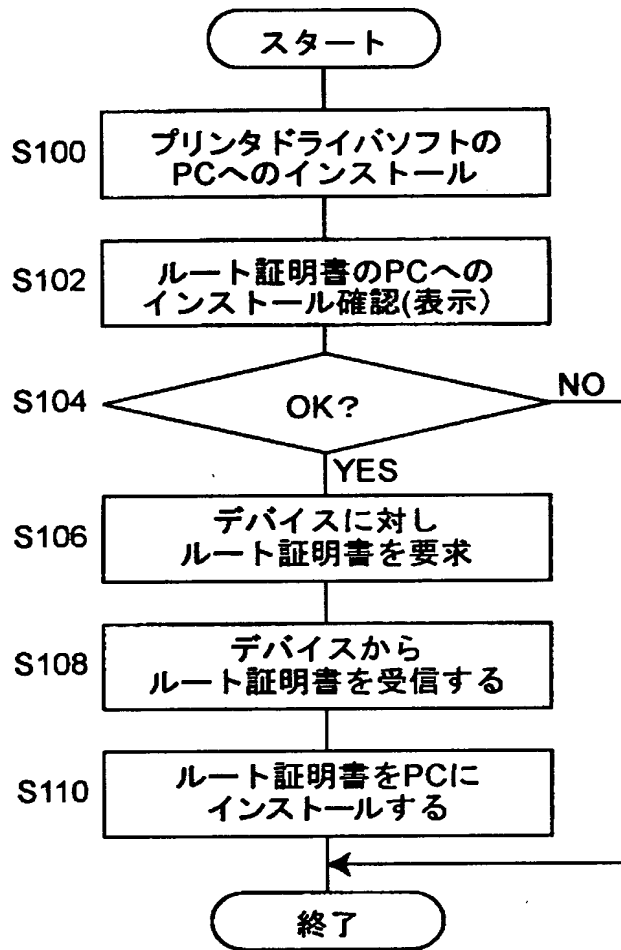
【図 8】



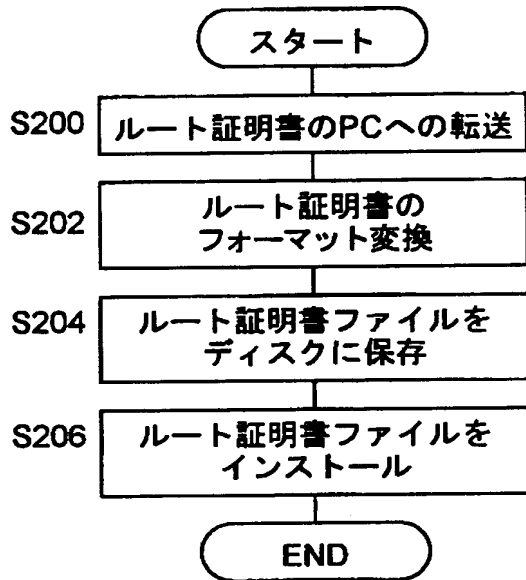
【図 9】



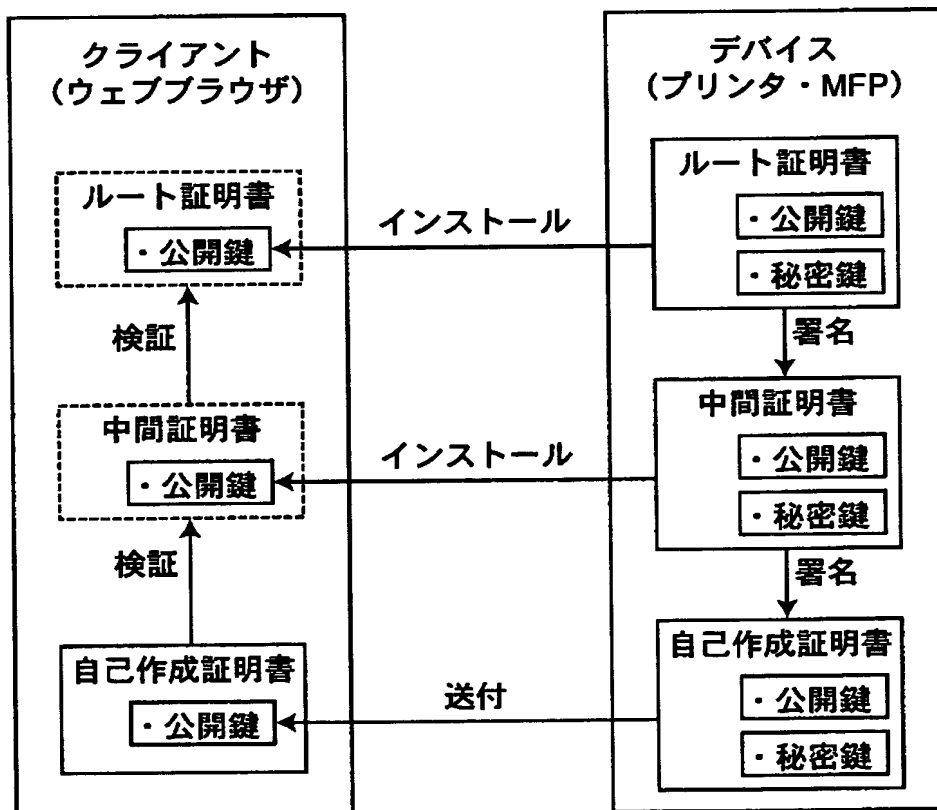
【図10】



【図 1 1】



【図 1 2】



【書類名】 要約書

【要約】

【課題】 デバイスとクライアントとがネットワークを介して通信するとき、ネットワークの外部から証明書を購入することなく、保証された証明書として使うことができる。

【解決手段】 デバイスとクライアントとがネットワークを介して通信する通信システムにおいて、デバイスは、生成された公開鍵と秘密鍵の対の中の公開鍵を含み、秘密鍵で署名されたルート証明書を保持する第1記憶手段と、ルート証明書を上位認証局として含む証明書として作成し、作成した証明書を秘密鍵を用いて署名する証明書作成手段と、証明書作成手段により作成された証明書をクライアントに送信する送信手段とを備える。また、クライアントは、前記の第1記憶手段に記憶されているルート証明書を記憶する第2記憶手段と、前記のデバイスから受け取った証明書の署名を公開鍵を用いて検証する検証手段とを備える。

【選択図】 図 5

出 願 人 履 歴 情 報

識別番号 [000006079]

1. 変更年月日 1994年 7月20日

[変更理由] 名称変更

住 所 大阪府大阪市中央区安土町二丁目3番13号 大阪国際ビル
氏 名 ミノルタ株式会社